AMENDMENT TO THE CLAIMS

1. (Currently amended) A computer-readable medium ~~containing~~ having stored computer

program code that when executed by a computer cause a computer system to recognize a

character based user interface having a plurality of host component types and to transform the

character based user interface to a web enabled user interface, the medium having code to control

the computer, the medium containing:

    code for scanning the character based user interface ~~by~~ for a plurality of agents;

    code in each agent for determining ~~which host component types exist in the character~~

~~based user interface, each agent determining~~ the existence of a different host component type

~~from the other agents~~ unique to the agent;

    code for defining a match region for each host component type found to exist by an agent

in the character based user interface;

    code for determining whether two or more match regions overlap; and

    code for rendering match regions associated with each agent to compose the web enabled

user interface.

    2. (currently amended) The computer-readable medium of claim 1 wherein the rendering

code further comprises:

    code rendering each match region as a widget, the aggregated widgets composing a

formatted output page.

3. (Previously amended) The computer-readable medium of claim 1 further comprising resolving code executed before the rendering code, comprising:

code for resolving a conflict between two or more match regions which overlap based on a policy to determine which agent associated with a match region controls the overlap region.

4. (Previously amended) The computer-readable medium of claim 3 wherein the resolving code comprises:

code for assigning a predetermined priority to each agent;

code for comparing the predetermined priority of the two or more conflicting agents; and

code for selecting the agent with the highest predetermined priority to control the overlapping region.

5. (Previously amended) The computer-readable medium of claim 3 wherein the resolving code further comprises:

code for comparing the size of the conflicting regions which overlap; and

code for selecting the agent having the smaller size region to control the overlapped region.

6. (Previously amended) The computer-readable medium of claim 3 wherein the resolving code further comprises:

code for assigning a dynamic priority to each conflicting region having a common overlapping region, the dynamic priority based on the projected amount of time expended to render each conflicting region; and

code for selecting the agent controlling the conflicting region having the highest priority to retain control over the overlapping region.

7. (Previously amended) The computer-readable medium of claim 4 further containing code for controlling the conflicting agents to negotiate whether to relinquish control of at least the overlap region.

8. (Previously amended) A computer system for recognizing a character based user interface having a plurality of host component types and transforming the character based user interface to a web enabled user interface, the computer system comprising:

a memory comprising a plurality of agent objects to scan the character based user interface, each agent object determining the existence of a different host component type from the other agents, each agent object defining a match region for each host component type found to exist in the character based user interface, each agent object rendering its associated match region to compose the web enabled user interface; and

a processor for running the plurality of agent objects.

9. (Currently amended) The computer system of claim 8 wherein each agent ~~rendering~~ renders each match region as a widget, the aggregated widgets composing a formatted output page.

10. (Currently amended) The computer system of claim 8 wherein the memory further ~~comprising~~ comprises:

an agent manager for determining whether two or more match regions overlap.

11. (Currently amended) The computer system of claim 10 wherein ~~two or more agents resolve~~ the system resolves a conflict between two or more overlapping match regions based on a policy to determine which agent associated with one match region controls the overlap region, the processor running the policy.

12. (original)   The computer system of claim 11 wherein the policy executed by the processor comprises:

assigning a predetermined priority to each agent;

comparing the predetermined priority of the two or more conflicting agents; and

selecting the agent with the highest predetermined priority to control the overlapping region.

13. (original)   The computer system of claim 11 wherein the policy executed by the

processor comprises:

comparing the size of the conflicting regions which overlap; and

selecting the agent having the smaller size region to control the overlapped region.

14. (original)   The computer system of claim 11 wherein the policy executed by the

processor comprises:

assigning a dynamic priority to each conflicting region having a common overlapping

region, the dynamic priority based on the projected amount of time expended to render each

conflicting region; and

selecting the agent controlling the conflicting region having the highest priority to retain

control over the overlapping region.

15. (original)   A method for recognizing a character based user interface having a

plurality of host component types and transforming the character based user interface to a web

enabled user interface, the method comprising:

scanning the character based user interface ~~by~~ <u>for</u> a plurality of agents;

determining which host component types exist in the character based user interface, each

agent determining the existence of a different host component type from the other agents;

defining a match region for each host component type found to exist by an agent in the

character based user interface;

determining whether two or more match regions overlap; and

rendering match regions associated with each agent to compose the web enabled user interface.

16. (Currently amended) The method of claim 15 wherein the render step further comprises the step of:

rendering each match region as a widget, the aggregated widgets composing a formatted output page.

17. (original)   The method of claim 15 further comprising a step before the rendering step, the step comprising:

resolving a conflict between two or more match regions which overlap based on a policy to determine which agent associated with a match region controls the overlap region.

18. (original)   The method of claim 17 wherein the policy comprises the steps of:

assigning a predetermined priority to each agent;

comparing the predetermined priority of the two or more conflicting agents; and

selecting the agent with the highest predetermined priority to control the overlapping region.

19. (original)   The method of claim 17 wherein the policy comprises the steps of:

comparing the size of the conflicting regions which overlap; and

selecting the agent having the smaller size region to control the overlapped region.

20. (original)   The method of claim 17 wherein the policy comprises the steps of:

assigning a dynamic priority to each conflicting region having a common overlapping region, the dynamic priority based on the projected amount of time expended to render each conflicting region; and

selecting the agent controlling the conflicting region having the highest priority to retain control over the overlapping region.

21. (original)   The method of claim 18 wherein the conflicting agents negotiate whether to relinquish control of at least the overlap region.